7 Steps to Mastering Data Preparation with Python

원문 : https://www.kdnuggets.com/2017/06/7-steps-mastering-data-preparation-python.html

Data preparation, Cleaning, Pre-processing, Cleansing, Wrangling 등에서 어떤 용어를 선택해도, Machine Learning, Big Data, Data mining, Data science 분야에서 크게 보면 데이터 활동을 사전 모델링하는 관련된 것이다.

어떤 용어를 선택하든, Machine Learning, Big Data, Data mining, Data science 커뮤니티에서 대략적으로 관련된 사전 모델링 데이터 활동 세트를 말합니다.

데이터 분석 모델

CRISP-DM model 은 공개 표준으로 데이터 마이닝 전문가에 의해 공통 관심사를 모델링하는 절차이다.

비교를 위해 두 가지 용어의 정의를 살펴보자,

데이터 클린징 (Data Cleansing)

..기록 세트, 테이블 또는 데이터베이스에서 손상되거나 부정확한 기록을 탐지 및 수정(또는 제거하는)하는 과정이며, **데이터의 불완전하거나, 부정확하거나, 관련 없는 부분을** 확인한 다음, 거친 데이터를 대체, 수정 또는 삭제하는 프로세스입니다. 데이터 정리는 데이터 분쇄 도구를 사용하거나 스크립팅을 통한 배치 처리로 대화형으로 수행될 수 있습니다.

데이터 충돌 (Data Wrangling)

...반자동화된 도구를 사용하여 데이터를 보다 편리하게 사용할 수 있도록 한 "원시" 형식의 데이터를 수동으로 변환하거나 매핑하는 프로세스입니다. 여기에는 추가 용도, 데이터 시각화, 데이터 통합, 통계 모델 교육 및 기타 많은 잠재적 용도가 포함될 수 있습니다. 일반적으로 데이터를 프로세스로 분리하는 작업은 데이터 원본에서 원시 형태로 데이터를 추출하는 것으로 시작하여, 알고리즘(예: 정렬)을 사용하여 원시 데이터를 "제거"하거나, 데이터 구조를 분석합니다.

처음에는 "data sink"라는 용어가 마음에 들지 않는다고 지적했지만, 계속해서 "데이터의 불완전하거나, 부정확하거나, 부정확하거나, 관련 없는 부분을 파악"한 다음, 데이터의 더러운 부분을 대체, 수정 또는 삭제하는 것이라고 말하고 싶습니다.." 데이터 준비를 포함하는 것으로 간주하는 "통계 모델 교육"에 이르기까지, 또는 "모델 구성까지 포함하되 포함하지는 않는 데이터 소싱으로부터의 모든 것"에 이르기까지. 그것은 우리가 앞으로 나아갈 애매모호한 정의이다.

Step1: 준비를 준비

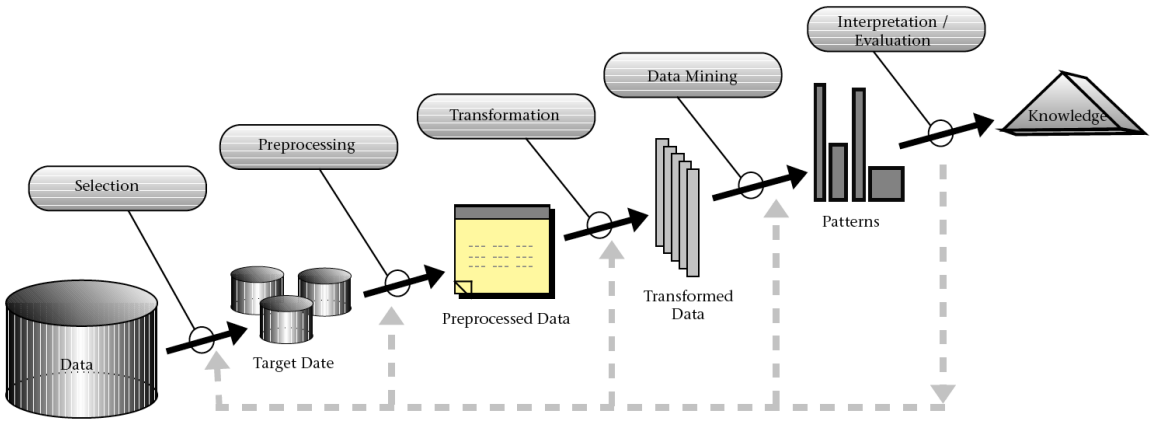
먼저, 다른 모든 사람들이 이미 말한 것을 강조해 봅시다: 이 데이터 준비 단계는 기계 학습 과제 이전의 예비 단계가 아니라 실제적으로 기계 학습의 중요한 요소(또는 심지어 다수)가 포함될 것이라고 주장될 수 있습니다. 그러나 이를 위해 데이터 준비는 모델링에서 자체적인 방법으로 분리할 것입니다.

Python이 생태계이기 때문에 우리가 다룰 대부분은 Python과 관련이 있을 것이다. 초보자에게 Pandas는 데이터 조작 및 분석 라이브러리이며 Python 과학 프로그래밍 스택의 기초 중 하나이며 데이터 준비와 관련된 많은 작업에 매우 적합합니다.

데이터 준비는 위에 나와 있는 *CRISP-DM* 모델에서 확인할 수 있습니다. 또한 데이터 준비와 KDD 프로세스의 프레임워크를 동일시 할 수 있습니다. 선택, 사전 처리 및 변환의 세 가지 주요 단계입니다. 이를 좀 더 세분화된 수준으로 나눌 수 있지만, 거시적 수준에서 *KDD* 프로세스 의 이러한 단계는 데이터 분리가 무엇인지 포괄합니다.

KDD Process

데이터 마이닝은 데이터베이스에서 지식 검색(Knowledge Discovery in Databases, or KDD) 이라고도 하는데, 이전에는 알 수 없었던 유용한 암묵적 정보를 데이터베이스에 저장된 데이터에서 부적절하게 추출하는 것을 의미합니다. 이런 절차를 아래 그림에서 알아보자,



- http://www2.cs.uregina.ca/~dbd/cs831/notes/kdd/1_kdd.html

- https://danrodgar.github.io/DASE/what-is-data-mining-knowledge-discovery-in-databases-kdd.html

1. Data Cleaning

Data cleaning is defined as removal of noisy and irrelevant data from collection. Cleaning in case of Missing values. Cleaning noisy data, where noise is a random or variance error. Cleaning with Data discrepancy detection and Data transformation tools.

2. Data Integration:

Data integration is defined as heterogeneous data from multiple sources combined in a common source(DataWarehouse). Data integration using Data Migration tools. Data integration using Data Synchronization tools. Data integration using ETL(Extract-Load-Transformation) process.

3. Data Selection:

Data selection is defined as the process where data relevant to the analysis is decided and retrieved from the data collection. Data selection using Neural network. Data selection using Decision Trees. Data selection using Naive bayes. Data selection using Clustering, Regression, etc.

4. Data Transformation:

Data Transformation is defined as the process of transforming data into appropriate form required by mining procedure. Data Transformation is a two step process:

- Data Mapping: Assigning elements from source base to destination to capture transformations.
- Code generation: Creation of the actual transformation program.

5. Data Mining:

Data mining is defined as clever techniques that are applied to extract patterns potentially useful. Transforms task relevant data into patterns. Decides purpose of model using classification or characterization.

6. Pattern Evaluation:

Pattern Evaluation is defined as as identifying strictly increasing patterns representing knowledge based on given measures. Find interestingness score of each pattern. Uses summarization and Visualization to make data understandable by user.

7. Knowledge representation:

Knowledge representation is defined as technique which utilizes visualization tools to represent data mining results. Generate reports. Generate tables. Generate discriminant rules, classification rules, characterization rules, etc.

> Note: KDD is an iterative process where evaluation measures can be enhanced, mining can be refined, new data can be integrated and transformed in order to

> get different and more appropriate results. Preprocessing of databases consists of Data cleaning and Data Integration.

Pandas 에 대해서 아래 정보가 도움이 될 것이다:

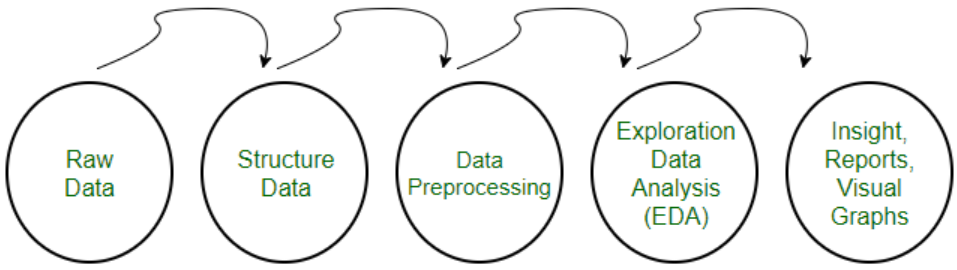
- 10 Minutes to pandas, Pandas documentation
- Intro to pandas data structures, by Greg Reda
- Modern Pandas (in 7 parts), by Tom Augspurger

다음 인터뷰 기사도 도움이 된다:

- Data Preparation Tips, Tricks, and Tools: An Interview with the Insiders, by Matthew Mayo

Step 2: 탐색적 데이터 분석(EDA)

탐색적 데이터 분석(Exploratory Data Analysis, EDA)은 더 큰 데이터 분석, 데이터 과학 또는 기계 학습 프로젝트의 필수적인 요소입니다. 데이터 작업을 수행하기 전에 데이터를 이해하는 것은 매우 좋은 아이디어일 뿐만 아니라 중요한 작업을 수행할 계획이라면 우선입니다.



Chloe Mawer는 탐색적 데이터 분석의 가치에서 다음과 같이 설명합니다.

> 전반적으로 EDA는 시각 및 정량적 방법을 사용하여 데이터셋을 이해하고 요약하는 관행입니다. 기계 학습 또는 통계적 모델링에 뛰어들기 전에 취하는 것은 중요한 단계입니다. 왜냐하면 그것은 당면한 문제에 적합한 모델을 개발하고 그 결과를 정확하게 해석하는 데 필요한 맥락을 제공하기 때문입니다.

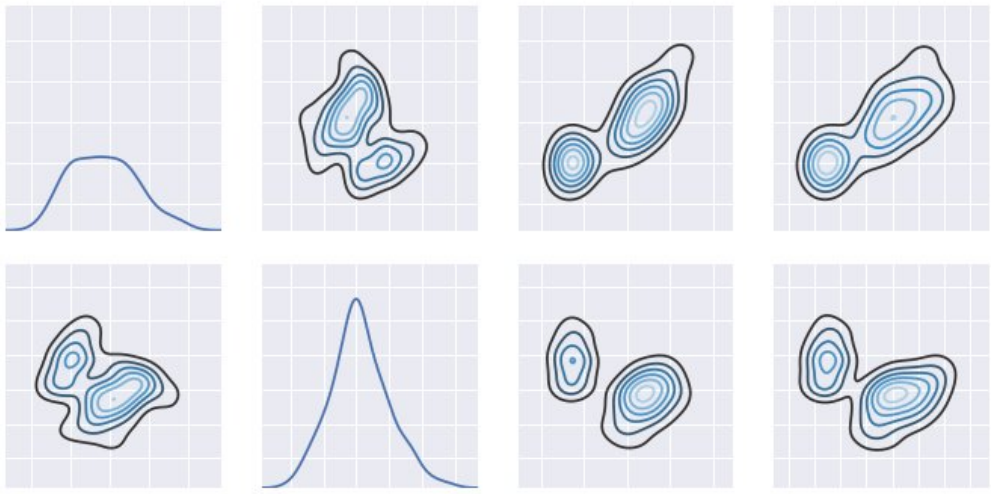
기본적인 요점은 예측 알고리즘을 효과적으로 선택하거나 데이터 준비의 나머지 단계를 계획하기 전에 먼저 **데이터의 구성**을 알아야 한다는 것입니다. 가장 뜨거운 알고리즘에 데이터셋을 던져 최고를 바라는 것은 전략이 아닙니다. 우리의 **요구조건**을 아는 것 또한 중요하다. 인간이 판독할 수 있는 출력이 높은 우선순위로 판단될 경우, 신경 네트워크를 사용하는 것이 그 결과를 감소시키지 않을 것이다. 의사결정 트리는 주어진 시나리오에서 가장 높은 분류 정확도를 제공하지 않을 수 있지만, 아마도 그러한 정확성의 희생은 해독 가능한 과정과 교환하여 허용될 것이다.

Chloe는 EDA가 일반적으로 다음 방법의 조합을 수반한다고 말합니:.

- 원시 데이터 집합의 각 필드에 대한 **요약 통계 및 일변량 시각화**
- 데이터 집합의 각 **변수와 관심 대상 변수 사이의 관계를 평가**하기 위한 치우침 시각화 및 요약 통계량(예: 변동될 때까지의 시간, 소비)
- 데이터의 여러 필드 간의 **상호 작용을 이해하기 위한 다변량 시각화**
- 치수 감소: **관측치 간 가장 큰 차이를 설명**하고 감소된 데이터 볼륨을 처리할 수 있는 데이터 필드를 이해합니다.

- 데이터를 몇 개의 작은 데이터 포인트로 분해함으로써 동작 패턴을 보다 쉽게 식별할 수 있는 차별화된 그룹으로 유사한 **관측치 클러스터링**

Chloe's article



[그림. 붓꽃 데이터세트 분포 가시화, https://en.wikipedia.org/wiki/Iris_flower_data_set]

타이타닉 데이터 세트에 적용된 탐색 데이터 분석 프로세스의 예는 다음을 참조하십시오:

- Titanic EDA, by Tarek Dib

To get an idea of using **Seaborn**, a statistical data visualization library, to visualize a dataset distribution see:

- Visualizing the distribution of a dataset, Seaborn documentation

A pair of additional libraries which may be useful for data exploration:

- Dora - Tools for exploratory data analysis in Python, by Nathan Epstein
- pandas-profiling - Create HTML profiling reports from pandas DataFrame objects, by Jos Polfliet

For a quick word on categorical data, see the following:

- Qualitative Research Methods for Data Science?, by Kevin Gray
- Generating a wordcloud in Python, by Andreas Mueller

Step 3: 결측치 다루기

손실된 데이터를 처리하기 위한 다양한 전략이 있으며, 이러한 전략 중 어느 것도 보편적으로 적용되지 않습니다.

일부 사람들은 "빈 값이 포함된 인스턴스를 사용하지 않는다"고 말할 것이다. 다른 사람들은 "결측값을 대체하기 위해 속성의 평균값을 절대 사용하지 않습니다."라고 주장합니다. 반대로 "데이터셋을 알려진 클래스 수로 먼저 클러스터링한 다음 클러스터 내 회귀 분석을 사용하여 결측값을 계산하는 등 도매로 승인된 더 복잡한 방법을 들을 수 있습니다.

몇몇 유형의 데이터 및 프로세스에서는 누락된 값을 처리하는 데 있어 서로 다른 모범 사례를 제안합니다. 그러나 이러한 유형의 지식은 경험 기반이며 영역을 기반으로 하기 때문에 우리는 채택

할 수 있는 보다 기본적인 전략에 초점을 맞출 것이다.

결측값을 처리하는 몇 가지 일반적인 방법은 다음과 같습니다:

- 삭제 사례
- 속성 삭제
- 누락된 모든 값에 대한 속성 평균 사용
- 누락된 모든 값에 대해 속성 중위수 적용
- 누락된 모든 값에 대한 속성 모드 사용
- 회귀 분석을 사용하여 결측값 귀속성

위에서 언급한 바와 같이, 채택되는 모델링 방법의 유형은 사용자의 결정에 영향을 미칩니다. 예를 들어, 의사결정 트리는 결측값을 수용할 수 없습니다. 또한 데이터 집합에서 결측값을 결정할 때 생각할 수 있는 통계 방법을 기술적으로 접할 수 있지만 나열된 접근 방식은 시도되고 테스트되며 일반적으로 사용되는 접근 방식을 사용할 수 있습니다.

Python 과 Pandas 기반 결측치 관련 :

- Working with missing data, Pandas documentation
- pandas.DataFrame.fillna, Pandas documentation

Pandas DataFrame에서 누락된 값을 교체하려는 항목으로 채울 수 있는 많은 방법이 있습니다. 다음은 몇 가지 기본적인 예입니다.

```
# Drop the columns where all elements are missing values:
df.dropna(axis=1, how='all')

# Drop the columns where any of the elements are missing values
df.dropna(axis=1, how='any')

# Keep only the rows which contain 2 missing values maximum
df.dropna(thresh=2)

# Drop the columns where any of the elements are missing values
df.dropna(axis=1, how='any')

# Fill all missing values with the mean of the particular column
df.fillna(df.mean())

# Fill any missing value in column 'A' with the column median
df['A'].fillna(df['A'].median())

# Fill any missing value in column 'Depeche' with the column mode
df['Depeche'].fillna(df['Depeche'].mode())
```

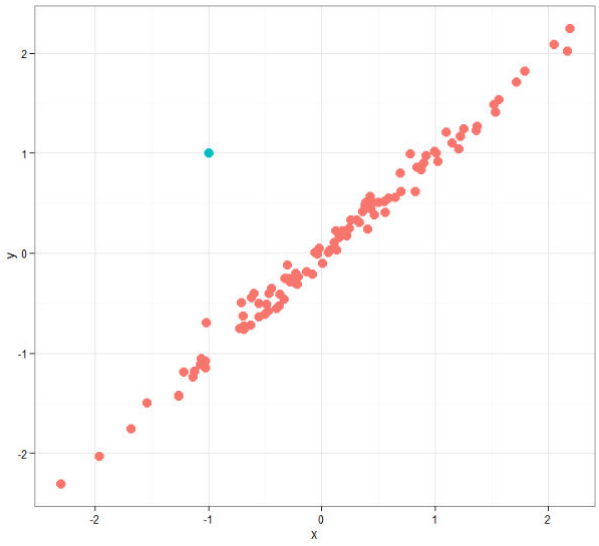
결측값, 특히 귀책점에 대한 일부 추가 보기는 다음을 참조하십시오.

- How to Treat Missing Values in Your Data: Part I, by Jacob Joseph
- How to Treat Missing Values in Your Data: Part II, by Jacob Joseph

Step 4: Dealing with Outliers

이 튜토리얼은 모델링을 할 때 데이터의 특이치를 처리하는 전략을 작성하는 것이 아닙니다. 특이치를 모델링에 포함시키는 것이 적절한 경우도 있고, 다른 사람이 말하지 않는 경우도 있습니다.

이것은 상황에 따라 다르며, 아무도 당신의 상황이 A열인지 B열인지에 대해 전면적으로 주장할 수 없다.



[특이치를 찾을 수 있나?]

특치치에 대한 의견:

- Outliers: To Drop or Not to Drop, The Analysis Factor
- Is it OK to remove outliers from data?, Stack Exchange

특이치는 잘못된 데이터 수집의 결과일 수도 있고, 실제로 양호하고 비정상적인 데이터일 수도 있습니다. 이러한 시나리오는 서로 다른 두 가지 시나리오이므로, 누락된 값을 처리하는 것과 유사한 "한 가지 크기"의 조언이 여기에 적용되지 않습니다. 위의 분석 요소 기사에서 특히 유용한 통찰력은 다음과 같습니다.

> 한 가지 옵션은 변환을 시도하는 것입니다. 제곱근과 로그 변환은 모두 높은 숫자를 끌어들입니다. 이렇게 하면 특이치가 종속 변수인 경우 가정이 더 잘 작동하고 특이치가 독립 변수인 경우 단일 점의 영향을 줄일 수 있습니다.

데이터셋에 특이치를 남겨둘지 여부를 결정하겠습니다. 그러나 모형의 경우 특이치 데이터를 처리해야 하는 몇 가지 방법이 나와 있습니다.

- 3 methods to deal with outliers, by Alberto Quesada
- Removing Outliers Using Standard Deviation in Python, by Punit Jajodia
- Remove Outliers in Pandas DataFrame using Percentiles, Stack Overflow

Step 5: 불균형 데이터 처리

그렇다면, 다른 강력한 데이터셋이 결측값과 특이치가 모두 없는 경우, 두 가지 클래스로 구성된다면 어떨까요? 하나는 인스턴스의 95%를 포함하는 것이고 다른 하나는 5%만 포함하는 것입니다. 아니면 더 나쁜 것은 -- 99.8 대 0.2 퍼센트?

만일 그렇다면, 최소한 클래스에 관한 한 데이터셋이 불균형한 것입니다. 이것은 제가 지적할 필요가 없는 방법으로 문제가 될 수 있습니다. 하지만 아직 데이터를 옆으로 던질 필요는 없습니다. 이 문제를 해결하기 위한 전략이 있습니다.

이러한 데이터 집합 특성은 실제로 데이터 준비 작업은 아니지만 이러한 데이터 집합 특성은 데이터 준비 단계(EDA의 중요성) 초기에 알려지며, 이러한 데이터의 유효성은 이 준비 단계에서 반드

시 즉시 평가할 수 있습니다.

먼저, Tom Fawcett의 접근 방법에 대한 토론을 살펴보십시오.

- Learning from Imbalanced Classes, by Tom Fawcett

Next, take a look at this discussion on techniques for handling class imbalance:

- 7 Techniques to Handle Imbalanced Data, by Ye Wu & Rick Radewagen

불균형 데이터에 도달할 수 있는 이유와 일부 영역에서 다른 영역보다 훨씬 자주 발생할 수 있는 이유에 대한 설명(위 링크된 7가지 기술에서 불균형 데이터 처리까지)

> 이러한 영역에서 사용되는 데이터는 종종 드물지만 "관심 있는" 이벤트(예: 신용 카드를 사용하는 사기범, 사용자 클릭 광고 또는 네트워크 손상 서버 검사)의 1% 미만입니다. 그러나 대부분의 기계 학습 알고리즘은 불균형 데이터 세트에서는 제대로 작동하지 않습니다. 다음의 7가지 기법은 비정상적인 클래스를 감지하도록 분류기를 훈련시키는 데 도움이 될 수 있습니다.

Step 6: Data Transformations

위키피이다에 데이터 변환)은:

> 통계에서 데이터 변환은 데이터 세트의 각 점에 결정론적 수학 함수를 적용하는 것이다. 즉, 각 데이터 지점 zi가 변환된 값 $yi = f(zi)$로 대체된다. 여기서 f는 함수이다. 일반적으로 변환은 데이터가 적용될 통계적 추론 절차의 가정을 더 가깝게 충족하거나 그래프의 해석 가능성과 모양을 개선하기 위해 적용된다.

데이터 변환은 데이터 준비의 가장 중요한 측면 중 하나이며, 대부분의 다른 측면보다 더 많은 정교함을 필요로 합니다. 결측값이 데이터에서 드러날 경우, 일반적으로 쉽게 찾을 수 있으며, 위에서 설명한 일반적인 방법 중 하나 또는 도메인의 통찰력을 통해 더 복잡한 조치를 다룰 수 있습니다. 그러나 데이터 변환이 필요한 시점과 경우(필요한 변환 유형은 말할 것도 없고)는 쉽게 식별할 수 없는 경우가 많습니다.

변형이 유용한 시기와 이유를 일반화하는 대신, 더 나은 제어력을 얻기 위해 몇 가지 구체적인 변환을 살펴보겠습니다.

Scikit-learn 설명서에서 이 개요는 가장 중요한 사전 처리 변환(예: 표준화, 표준화 및 이중화)에 대한 몇 가지 근거를 제공합니다(몇 가지 다른 변환도 함께 제공).

Preprocessing data, Scikit-learn 참조

단일 열 인코딩 "분류 및 회귀 알고리즘에 더 적합한 형식으로 범주형 특징을 변환한다"(아래의 첫 번째 링크에서 발췌). 팬더를 사용한 접근은 물론 아래 1회성 변환에 대한 논의를 참조하십시오.

- What is one hot encoding and when is it used in data science?, Quora answer by Håkon Hapnes Strand
- How can I one hot encode in Python?, Stack Overflow

로그 분포 변환은 "비선형적이지만 선형 모형으로 변환할 수 있는 모델"인 경우 유용합니다(아래에서). 아래에서 제대로 인식되지 않는 변환 유형에 대해 자세히 알아보십시오.

- When (and why) should you take the log of a distribution (of numbers)?, Stack Exchange

위에서 설명한 것처럼 데이터와 요구 사항에 따라 다양한 변환이 가능합니다. 향후 데이터 변혁에 대해 자세히 알아보고, 그 시간 동안 보다 심도 있는 논의를 남기고 싶습니다.

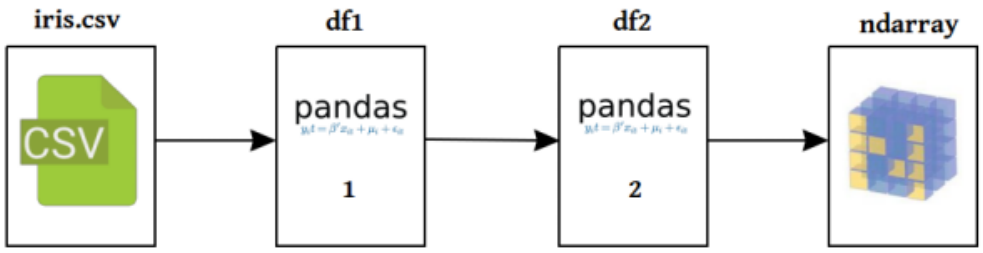
> 이 전체 논의는 또한 특정한 이유로 특징 선택에 대한 언급은 완전히 의도적으로 생략하고 있다는 점에 유의하십시오. 이것은 훨씬 더 광범위한 논의에서 단순한 몇 문장 이상의 가치가 있습니다. 기능 선택을 위한 유사한 지침이 마련될 예정이며, 완료되면 여기에 연결됩니다.

Step 7: Finishing Touches & Moving Ahead

그러시죠. 데이터가 "세정"되어 있습니다. 이는 귀하가 현재 유효한 Pandas DataFrame을 보유하고 있음을 의미합니다. 하지만 당신은 그것으로 무엇을 하나요?

모델을 구축하기 위해 데이터를 기계 학습 알고리즘에 바로 공급하려면 데이터가 더 적절한 표현으로 필요할 수 있습니다. Python 생태계에서, 그것은 일반적으로 무미건조한 ndarray이다. 그곳에 가는 것에 대한 몇 가지 예비 아이디어를 위해 다음을 살펴볼 수 있습니다.

- Turning a Pandas Dataframe to an array and evaluate Multiple Linear Regression Model, Stack Overflow



[그림. 간단한 데이터 준비 처리]

Python에서 기계 학습을 위한 적절한 표현으로 정리된 데이터를 가지고 있다면, 지금 바로 사용할 준비가 된 바로 그 장을 확인해 보는 것이 어떨까요?

- 7 Steps to Mastering Machine Learning With Python, by Matthew Mayo
- 7 More Steps to Mastering Machine Learning With Python, by Matthew Mayo

아직 모델링을 하고 싶지 않다면요? 그렇다면 이 데이터를 어떤 스토리지 형태로 출력하고자 한다면 어떻게 하시겠습니까? 다음은 Pandas DataFrame 스토리지에 대한 몇 가지 정보입니다.

- Writing a Pandas DataFrame to MySQL, Stack Overflow
- Quick HDF5 with Pandas, by Giuseppe Vettigli

모든 종류의 기계 학습 작업에 적용되는 프로세스인 교육 및 테스트 세트로 데이터 세트를 분할하는 것을 포함하여 앞으로 나아가기 전에 데이터셋과 관련된 추가 고려 사항이 있다는 점을 잊지 마십시오.

- Numpy: How to split/partition a dataset (array) into training and test datasets for, e.g., cross validation?, Stack Overflow
- Is there a Python function that splits data into train, cross validation and test sets?, Quora answer by Harizo Rajaona

그리고 순수한 종결로서, 다음은 데이터 준비에 대한 몇 가지 추가 사항입니다.

- Tidying Data in Python, by Jean-Nicholas Hould
- Doing Data Science: A Kaggle Walkthrough Part 3 – Cleaning Data, by Brett Romero
- Machine Learning Workflows in Python from Scratch Part 1: Data Preparation, by Matthew Mayo